79МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное образовательное
учреждение высшего образования
ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

Институт математики, механики и компьютерных наук
имени И. И. Воровича

ОТЧЕТ

БЛОЧНЫЕ ВЫЧИСЛЕНИЯ. МОДЕЛИ ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРОГРАММ. БЛОЧНЫЕ РАЗМЕЩЕНИЯ МАССИВОВ, ДОПОЛНЯЮЩИЕ БЛОЧНЫЕ ВЫЧИСЛЕНИЯ

Студентки 4 курса
2 группы
А. В. Домбровской

2018 г.

Задание 30.

Написать программу блочного умножения двух матриц C = A*B.

Матрица A симметричная, хранится как верхне-треугольная. Хранится в виде одномерного массива по блочным строкам.

Матрица B верхне-треугольная. Хранится в виде одномерного массива по блочным столбцам.

Распараллелить блочную программу умножения двух матриц C = A*B с использованием технологии OpenMP двумя способами

- Перемножение каждых двух блоков выполнить параллельно
- В разных вычислительных ядрах одновременно перемножать разные пары блоков.

Определить оптимальные размеры блоков в обоих случаях.

Провести численные эксперименты и построить таблицу сравнений времени выполнения различных программных реализаций решения задачи. Определить лучшие реализации.

Проверить корректность (правильность) программ.

Алгоритм перемножения матриц по блокам в общем виде

Перемножение матриц разбивается на 3 общих цикла, как в стандартном способе, но внутри третьего цикла появляется проверка на номер блока, по которому определяется его расположение в матрице.

Если для блока матрицы A номер определяет его положение на диагонали матрицы, то этот блок берется с элементами ниже диагонали, равными элементам лежащим симметрично им относительно диагонали, поскольку A хранится как верхнее-треугольная матрица.

Если же номер блока соответствует блоку, лежащему выше диагонали, то этот блок берется без изменений.

Если – ниже диагонали, то блок определяется симметричным относительно главной диагонали блоком, но в измененном виде, а именно – транспонированном.

Для элементов матрицы B таких действий проводить не требуется, поскольку она верхне-треугольная.

Внутри цикла по k нам придется брать элементы из векторов, записанных по блокам, со смещением на необходимое количество позиций. Поскольку матрица A записана построчно, а матрица B – по столбцам, то алгоритм их перемножения выглядит достаточно просто:

1) Для начала берем первые n элементов из вектора A по размеру блока, перемножаем их с первыми n элементами из вектора B, как обычно делаем это в матрицах;
2) Затем перемножаем те же самые элементы из A с элементами из следующего столбца B, а то есть, к начальному индексу прибавляем размер блока;
3) Переходим на следующую строку матрицы A, то есть к вектору A прибавляем размер блока, и точно так же умножаем на элементы из B;
4) Такую процедуру повторяем для всех блоков матрицы.

Элементы матрицы C хранятся построчно.

Проверка правильности выполнения программы на примере матриц (4x4)

Начальные данные:

```
Matrix sizes (4x4):
 Matrix A:
1 7 4 0
7 9 4 8
4 4 8 2
0 8 2 4

 Matrix B:
5 5 1 7
0 1 1 5
0 0 2 7
0 0 0 6
```

Данные результатов для блоков размеров 1, 2 и 4:

```
block size: 1
Matrix A        1 7 4 0 9 4 8 8 2 4
Matrix B        5 5 1 7 1 1 5 2 7 6

 NONPARALLEL
For block size  1:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78

 PARALLEL THREADS
For block size  1:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78

 PARALLEL BLOCK
For block size  1:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78


block size: 2
Matrix A        1 7 7 9 4 0 4 8 8 2 2 4
Matrix B        5 0 5 1 1 1 7 5 2 0 7 6

 NONPARALLEL
For block size  2:    5  12  35  44  16  70  24 170  20  24   0   8  24 116  12  78

 PARALLEL THREADS
For block size  2:    5  12  35  44  16  70  24 170  20  24   0   8  24 116  12  78

 PARALLEL BLOCK
For block size  2:    5  12  35  44  16  70  24 170  20  24   0   8  24 116  12  78


block size: 4
Matrix A        1 7 4 0 7 9 4 8 4 4 8 2 0 8 2 4
Matrix B        5 0 0 0 5 1 0 0 1 1 2 0 7 5 7 6

 NONPARALLEL
For block size  4:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78

 PARALLEL THREADS
For block size  4:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78

 PARALLEL BLOCK
For block size  4:    5  12  16  70  35  44  24 170  20  24  24 116   0   8  12  78
```

Проверим корректность в среде Maple:

$$A := \begin{bmatrix} 1 & 7 & 4 & 0 \\ 7 & 9 & 4 & 8 \\ 4 & 4 & 8 & 2 \\ 0 & 8 & 2 & 4 \end{bmatrix}$$

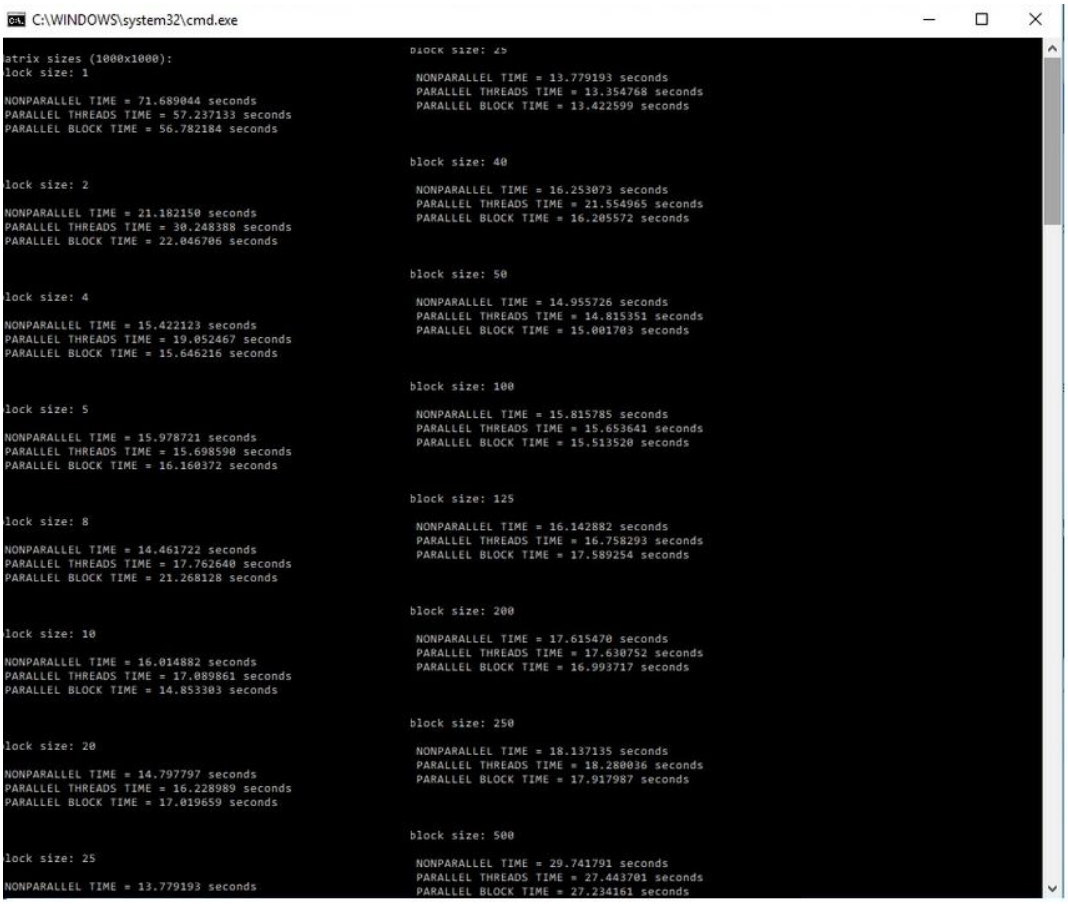
$$B := \begin{bmatrix} 5 & 5 & 1 & 7 \\ 0 & 1 & 1 & 5 \\ 0 & 0 & 2 & 7 \\ 0 & 0 & 0 & 6 \end{bmatrix}$$

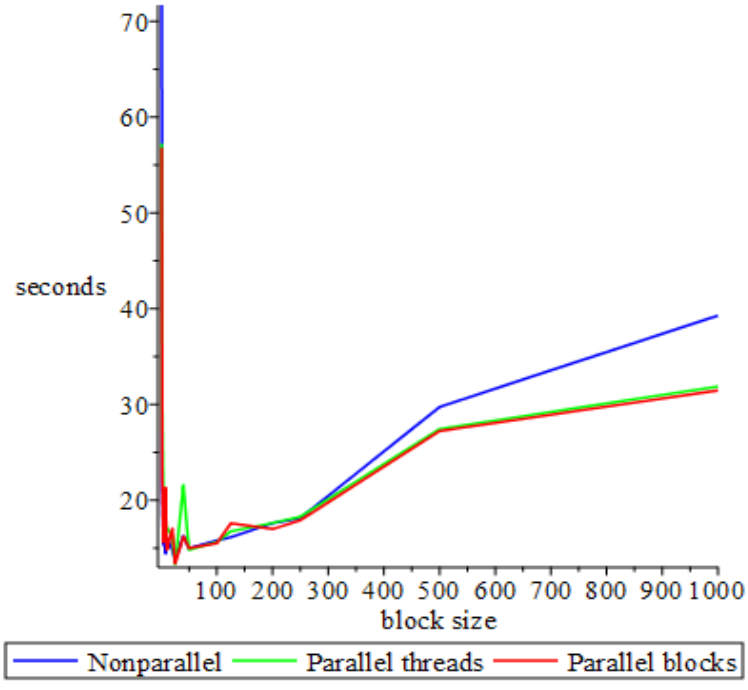
$$C := \begin{bmatrix} 5 & 12 & 16 & 70 \\ 35 & 44 & 24 & 170 \\ 20 & 24 & 24 & 116 \\ 0 & 8 & 12 & 78 \end{bmatrix} \quad (1)$$

Всё верно, значит теперь можно производить вычисления для замеров времени выполнения на больших значениях N – размерах матриц A и B.

Время выполнения для разных параметров

Вычисления производились для матриц размеров(1000x1000) и распараллеливании на 4 потока.





Характеристики устройства, на котором производились вычисления:

Процессор: Intel® Pentium® CPU N3540 @ 2.16GHz 2.16 GHz

Базовая скорость: 2.16 ГГц

Установленная память: 4.00 ГБ

Тип системы: 64-разрядная операционная система, процессор x64

Количество ядер: 4 ядра

Кэш L2: 2.0 МБ

Вывод

Наилучшим способом перемножения матриц является перемножение разных пар блоков параллельно в разных вычислительных ядрах. Большего всего разница заметна при разбиении на блоки размером по 1, но это одновременно является наиболее затратным по времени разбиением. Помимо этого разбиения, лучше всего заметна разница при разбиении на блоки размером (500x500) и (1000x1000).

Таким образом, наилучшим способом задача решается при максимальном разбиении и распараллеливании перемножения разных пар блоков на вычислительные ядра. Данные результаты полностью зависят от характеристик вычислительного устройства.